UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/752,172	12/29/2000	Patrick J. Shaughnessy	LOT9-2000-0024 US1	8918

27085 7590 04/05/2004

IBM CORPORATION
LOTUS SOFTWARE
ONE ROGERS STREET
CAMBRIDGE, MA 02142

EXAMINER	
BLACKWELL, JAMES H	
ART UNIT	**PAPER NUMBER**
2176	6

DATE MAILED: 04/05/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 09/752,172	**Applicant(s)** SHAUGHNESSY ET AL.	
	Examiner James H Blackwell	**Art Unit** 2176	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *29 December 2000*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-21* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-21* is/are rejected.

7)☒ Claim(s) *1,3,5,13 and 16* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *29 December 2000* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date 5.

4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application (PTO-152)

6) ☐ Other: _____.

DETAILED ACTION

Specification

1. The disclosure is objected to because of the following informalities: in Cross Reference to Related Applications section, entries left blank need to be updated. Likewise, in section dealing with copending, previously files applications. Appropriate correction is required.

Claim Rejections - 35 USC § 112

2. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

Claims 1-3, 5-7, 13, and 16 are rejected because of inappropriate use of Trademark names. Applicant is referred to MPEP, Section 2173.05(u) which is listed below:

> **2173.05(u) Trademarks or Trade Names in a Claim**
>
> The presence of a trademark or trade name in a claim is not, *per se*, improper under 35 U.S.C. 112, second paragraph, but the claim should be carefully analyzed to determine how the mark or name is used in the claim. It is important to recognize that a trademark or trade name is used to identify a source of goods, and not the goods themselves. Thus a trademark or trade name does not identify or describe the goods associated with the trademark or trade name. See definitions of trademark and trade name in MPEP § 608.01(v). A list of some trademarks is found in Appendix I.
>
> If the trademark or trade name is used in a claim as a limitation to identify or describe a particular material or product, the claim does not comply with the requirements of the 35 U.S.C. 112, second paragraph. *Ex parte Simpson*, 218 USPQ 1020 (Bd. App. 1982). The claim scope is uncertain since the trademark or trade name cannot be used properly to identify any particular material or product. In fact, the value of a trademark would be lost to the extent that it became descriptive of a product, rather than used as an identification of a source or origin of a product. Thus, the use of a trademark or trade name in a claim to identify or describe a material or product would not only render a claim indefinite, but would also constitute an improper use of the trademark or trade name.
>
> If a trademark or trade name appears in a claim and is not intended as a limitation in the claim, the question of why it is in the claim should be addressed. Does its presence in the claim cause confusion as to the scope of the claim? If so, the claim should be rejected under 35 U.S.C. 112, second paragraph.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 1-3 are rejected under 35 U.S.C. 103(a) as being unpatentable over Courtner et al. (hereinafter Courtner, Mastering Microsoft Office 2000 Professional Edition, Copyright 1999).

In regard to independent Claim 1,Courtner teaches adding an ftp location to a list of places accessible to place Microsoft or other documents in a location where other people can access the documents (p. 874 paragraphs 3-4; compare to Claim 1, ***"Method for using Microsoft Office documents in a place in collaboration space, comprising ..."***). Courtner also teaches creating a new Office Document by opening the New Office Document dialog box (see Fig. 2.1). Select Document Type and then click OK to open the application and begin creating a new document (pp. 23-24; compare to Claim 1, ***"... creating a document file using Microsoft Office"***). Courtner also teaches saving documents as Web pages using the Save As Web Page option on the File menu (pp. 950-951 paragraphs 1-6; compare to Claim 1, ***"... converting said document to a hypertext markup language (html) file"***). Courtner also teaches adding web folders to a web server into which HTML and other files can be saved (pp. 873-874; compare to Claim 1, ***"... saving said document file and said html file on a page in collaboration space"***). Courtner does not specifically teach ... *with said html* *file displayed in read mode and said document file displayed in edit mode*. However, it would have been obvious to one of ordinary skill in the art at the time of invention to have placed each of the two file types in said modes because it would have allowed for a side-by-side comparison of the formatted file (html file in read mode) and the source (document file in edit mode).

In regard to dependent Claim 2, Courtner teaches creating a template for a form. To create a new template file, (1) Choose File > New, highlight blank Document on the General page, select Template in the Create New Area, and then choose OK in the New dialog box. (2) Click on the Save button on the Standard toolbar to display the Save As dialog box. (3) Choose the folder in the Templates folder in which to store the template. Alternatively, create a new folder in the Templates folder. (4) Save the template to a named file with a .DOT file extension (p. 227). Courtner also teaches inserting the text and other objects that will be used in the form (p. 227-237). Courtner also teaches that if one wants to create a presentation jointly with one or more people in a workgroup, one may wish to keep the presentation in a shared network folder. That way, each collaborator can access and edit the same file rather than having multiple unfinished drafts (p. 359); compare to Claim 2, ***"... at said browser, dragging and dropping said document file into said place as a template file"*** and ***"creating from said template file a form"***). Courtner does not explicitly teach *dragging a dropping a document file into said place*. However Courtner does teach the drag and drop technique as a way to move and copy text short distances (p. 55). Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to have used drag and drop providing the benefit of not having had to use the two step process of cut and paste to move a document file from one place to another.

In regard to dependent Claim 3, Courtner teaches using a form to create a new Microsoft Word document (see Fig. 10.7; compare to Claim 3, ***"... using said form to create a new Microsoft document"***).

Claims 4-7, 17, 19, and 21 are rejected under 35 U.S.C. 103(a) as being unpatentable over Courtner in view of Salas et al. (hereinafter Salas, U.S. Patent No. 6,233,600) and in further view of Blumberg (U.S. Patent No. 6,708,309).

In regard to independent Claim 4 (and similarly independent Claim 21), Courtner teaches adding a form to any web page using Microsoft Word. This is done by adding a form control from the Web Tools toolbar (see pp. 1048-1056; compare to Claim 4 (and similarly Claim 21), ***"... creating a document file including web form fields"***). Courtner also teaches saving the document file (p. 1050; compare to Claim 4 (and similarly Claim 21), ***"... saving said document file"***). Courtner does not specifically teach *importing said document file to said place*. However, Salas teaches in Fig. 7, that a user creates a new file or modifies a file downloaded from the server (14) (step 702). Once the user is finished editing the file, it may be uploaded to the server (14) to allow other users access to it. The user signals that the file should be transmitted to the server (14) by dragging the file onto an eRoom displayed by the browser (step 704). Dropping the file into the displayed eRoom invokes an ActiveX control or a background daemon process that manages the upload of the file to the server (14) (Col. 13, lines

14-26; compare with Claim 4 (and similarly Claim 21), ***"... importing said document file to said place"***). Salas fails to teach ... *converting said document into a web file containing hyperlinks that point to web pages.* However, Blumberg teaches that documents archived on a server computer can be accessed by client computers by means of web browsers. If a document is not in hypertext markup language (HTML) format, additional server-side or client-side processing may be necessary. Such server-side processing involves auxiliary software on the server for converting each document page requested into HTML format for viewing by the client (Col. 6, lines 1-7; compare to Claim 4 (and similarly Claim 21), ***"... converting said document into a web file containing hyperlinks that point to web pages"***). It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Courtner, Salas, and Blumberg providing the benefit of having allowed an author to collaborate on a document with the option of having previewed it from a browser.

In regard to dependent Claim 5, Courtner teaches saving Office 2000, specifically Microsoft Word, documents as Web pages using the Save As Web Page option on the file menu. This saves an HTML version of the Word document (p. 950; compare to Claim 5, ***"... said document being a Microsoft Office document and said web file being a hypertext markup language (html) file"***).

In regard to dependent Claim 6, Courtner teaches creating web-based forms using the Web Tools toolbar (pp. 1048-1049; compare to Claim 6, ***"... including web form fields in said document file by selecting field icons from a web tools toolbar"***).

In regard to dependent Claim 7, Courtner teaches the Web Tools Toolbar containing checkbox control, option button control, drop-down boxes control, list boxes control, text boxes control, text areas control, submit buttons, submit with image, reset, hidden, and password control (pp. 1049-1050 compare to Claim 7, ***"... said toolbar including checkbox control, option or radio button control, drop down box control, listbox control, textbox control, text area control, submit control, submit with image control, reset control, hidden control, and password control"***).

In regard to independent Claim 17, Courtner teaches adding a form to any web page using Microsoft Word. This is done by adding a form control from the Web Tools toolbar (see pp. 1048-1056; compare to Claim 17, ***"... an editor for creating a document file including web form fields"***). Courtner fails to explicitly teach *a storage device for saving said document file*. However, Salas teaches a client workstation (12') using a local database (20') to store data objects related to a project, while external files related to a project (such as word processing document of a set of meeting minutes are stored in the client workstation's (12') local file system (Col. 4, lines 22-26; compare to claim 17, ***"... a storage device for saving said document file"***). Salas also teaches in Fig. 7, that a user creates a new file or modifies a file downloaded from the server (14) (step 702). Once the user is finished editing the file, it may be uploaded to the server (14) to allow other users access to it. The user signals that the file should be transmitted to the server (14) by dragging the file onto an eRoom displayed by the browser (step 704). Dropping the file into the displayed eRoom invokes an ActiveX control or a background daemon process that manages the upload of the file to the server (14) (Col. 13, lines 14-26; compare with Claim 17, ***"... an upload control for importing said document file to said place"***). Salas fails to teach *a server upon importing said document into said place, for converting said document into a web file containing hyperlinks that point to web pages*. However, Blumberg teaches that documents archived on a server computer can be accessed by client computers by means of web browsers. If a document is not in hypertext markup language (HTML) format, additional server-side or client-side processing may be necessary. Such server-side processing involves auxiliary software on the server for converting each document page requested into HTML format for viewing by the client (Col. 6, lines 1-7; compare to Claim 17, ***"... a server upon importing said document into said place, for converting said document into a web file containing hyperlinks that point to web pages"***). It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Courtner, Salas, and Blumberg providing the benefit of having allowed an author to collaborate on a document with the option of having first previewed it from a browser.

In regard to independent Claim 19, Courtner teaches adding a form to any web page using Microsoft Word. This is done by adding a form control from the Web Tools toolbar (see pp. 1048-1056; compare to Claim 19, ***"... creating a document file including web form fields"***). Courtner also teaches adding web folders to a web server into which HTML and other files can be saved (pp. 873-874; compare to Claim 19, ***"... saving said document file***"). Courtner fails to teach *importing said document file to said space*. However, Salas teaches in Fig. 7, that a user creates a new file or modifies a file downloaded from the server (14) (step 702). Once the user is finished editing the file, it may be uploaded to the server (14) to allow other users access to it. The user signals that the file should be transmitted to the server (14) by dragging the file onto an eRoom displayed by the browser (step 704). Dropping the file into the displayed eRoom invokes an ActiveX control or a background daemon process that manages the upload of the file to the server (14) (Col. 13, lines 14-26; compare with Claim 19,***"... importing said document file to said space"***). Salas fails to teach that *upon importing said document into said place, converting said document into a web file containing hyperlinks that point to web pages*. However, Blumberg teaches that documents archived on a server computer can be accessed by client computers by means of web browsers. If a document is not in hypertext markup language (HTML) format, additional server-side or client-side processing may be necessary. Such server-side processing involves auxiliary software on the server for converting each document page requested into HTML format for viewing by the client (Col. 6, lines 1-7; compare to Claim 19, ***"... upon importing said document into said place, converting said document into a web file containing hyperlinks that point to web pages"***). It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Courtner, Salas, and Blumberg providing the benefit of having allowed an author to have collaborated on a document with the option of having first previewed it from a browser.

Claims 8-9, and 20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Salas in view of Blumberg.

In regard to independent Claim 8, Salas teaches in Fig. 7, that a user creates a new file or modifies a file downloaded from the server (14) (step 702). Once the user is finished editing the file, it may be uploaded to the server (14) to allow other users access to it. The user signals that the file should be transmitted to the server (14) by dragging the file onto an eRoom displayed by the browser (step 704). Dropping the file into the displayed eRoom invokes an ActiveX control or a background daemon process which manages the upload of the file to the server (14) (Col. 13, lines 14-26; compare with Claim 8, ***"... presenting at a browser an interface window to said place including a new selector; responsive to user selection of said new selector and an import option, displaying an import scene with an empty upload control; dragging and dropping said document file to said upload control"***). Salas fails to teach that *responsive to receiving said document file in said upload control, converting said document file to hypertext files; displaying a document file icon in said upload control; importing said hypertext files into said place*. However, Blumberg teaches that documents archived on a server computer can be accessed by client computers by means of web browsers. If a document is not in hypertext markup language (HTML) format, additional server-side or client-side processing may be necessary. Such server-side processing involves auxiliary software on the server for converting each document page requested into HTML format for viewing by the client (Col. 6, lines 1-7; compare to Claim 8, ***"... responsive to receiving said document file in said upload control,***

converting said document file to hypertext files; displaying a document file icon in said upload control; importing said hypertext files into said place"). It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Salas and Blumberg providing the benefit of having a convenient user interface for uploading documents and having made them available via a browser for preview by collaborators.

In regard to dependent Claim 9, Salas teaches in Fig. 7, that a user creates a new file or modifies a file downloaded from the server (14) (step 702). Once the user is finished editing the file, it may be uploaded to the server (14) to allow other users access to it. The user signals that the file should be transmitted to the server (14) by dragging the file onto an eRoom displayed by the browser (step 704). Dropping the file into the displayed eRoom invokes an ActiveX control or a background daemon process that manages the upload of the file to the server (14) (Col. 13, lines 14-26; compare with Claim 9, ***"... responsive to said user publishing said document file to said collaboration space, uploading to a collaboration space server said document file and said hypertext files"***).

In regard to independent Claim 20, Salas teaches in Fig. 7, that a user creates a new file or modifies a file downloaded from the server (14) (step 702). Once the user is finished editing the file, it may be uploaded to the server (14) to allow other users access to it. The user signals that the file should be transmitted to the server (14) by dragging the file onto an eRoom displayed by the browser (step 704). Dropping the file into the displayed eRoom invokes an ActiveX control or a background daemon process that manages the upload of the file to the server (14) (Col. 13, lines 14-26; compare with Claim 20, ***"... presenting at a browser an interface window to said place including a new selector; responsive to a user selection of said new selector and an import option, displaying an import scene with an empty upload control; dragging and dropping said document file to said upload control"***). Salas fails to teach that *responsive to receiving said document file in said upload control, converting said document file to hypertext files; displaying a document file icon in said upload control importing said files into said place*. However, Blumberg teaches that documents archived on a server computer can be accessed by client computers by means of web browsers. If a document is not in hypertext markup language (HTML) format, additional server-side or client-side processing may be necessary. Such server-side processing involves auxiliary software on the server for converting each document page requested into HTML format for viewing by the client (Col. 6, lines 1-7; compare to Claim 20, ***"... responsive to receiving said document file in said upload control, converting said document file to hypertext files; displaying a document file icon in said upload control importing said files into said place"***). It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Salas and Blumberg providing the benefit of having allowed an author to have collaborated on a document with the option of having first previewed it from a browser.

Claims 10-16 are rejected under 35 U.S.C. 103(a) as being unpatentable over Salas in view of Blumberg and in further view of Courtner.

In regard to dependent Claim 10, Salas fails to teach that *said document file being a presentation file comprising a plurality of slides*. However, Courtner teaches that every PowerPoint presentation consists of a series of slides (p. 288; compare to Claim 10, ***"... said document file being a presentation file comprising a plurality of slides"***). It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Salas, Blumberg, and Courtner providing the benefit of having a convenient way of outlining information content.

In regard to dependent Claim 11, Salas fails to teach *displaying said slides in a same page together with previous, next, and zoom buttons*. However, Courtner teaches Viewing Slides. Specifically, slide view mode, which lets you work on one slide at a time. The main part of the tri-pane window displays the selected slide, while the left side window shows numbered slide icons with the selected slide shaded gray. If you wish to advance to the next or previous slide, you can use the buttons located at the bottom of the vertical scroll bar (p. 294). Courtner also teaches a zoom button (p. 58 for example of it in Microsoft Word, but exists in PowerPoint as well). Compare to Claim 11, ***"... displaying said slides in a same page together with previous, next, and zoom buttons"***. It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Salas, Blumberg, and Courtner providing the benefit of having had a convenient way to view and organize slides.

In regard to dependent Claim 12, Salas fails to teach that *responsive to selection of said zoom button, displaying a larger version of a current slide also including next and previous buttons*. However, Courtner teaches Viewing Slides. Specifically, slide view mode that lets you work on one slide at a time. The main part of the tri-pane window displays the selected slide, while the left side window shows numbered slide icons with the selected slide shaded gray. If you wish to advance to the next or previous slide, you can use the buttons located at the bottom of the vertical scroll bar (p. 294). Courtner also teaches a zoom button (p. 58 for example of it in Microsoft Word, but exists in PowerPoint as well). Compare to Claim 12, ***"... responsive to selection of said zoom button, displaying a larger version of a current slide also including next and previous buttons"***. It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Salas, Blumberg, and Courtner providing the benefit of having had a convenient way to view individual slides.

In regard to dependent Claim 13, Salas fails to teach that *said presentation file being a Microsoft PowerPoint file*. However, Courtner teaches that every PowerPoint presentation consists of a series of slides (p. 288; compare to Claim 13, ***"... said presentation file being a Microsoft PowerPoint file"***). Courtner also teaches saving Office 2000, which includes Microsoft PowerPoint, documents as Web pages using the Save As Web Page option on the file menu. This saves an HTML version of the PowerPoint presentation (p. 950; compare to Claim 13, ***"... said hypertext files being hypertext markup language (html) files"***). It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Salas, Blumberg, and Courtner providing the benefit of having had a specific application to produce a presentation based on slides and a means to convert the presentation to html format.

In regard to dependent Claim 14, Salas fails to teach *preparing and importing into a common page at said place a separate hypertext file for each said data object, together with any included images and support files, modifying said hypertext files to link to next and previous data objects within said document file; at read time, responsive to user selection of said next or previous control, issuing a uniform resource locator (url) to load and display a respective next or previous data object in sequence*. However, Courtner teaches that when you save a file as a web page, all supporting files, such as those for bullets, background textures, and graphics are saved in a separate subfolder. The subfolder is automatically named with the page name, followed by an underscore and the word files; for example, My Web Page_files. If you move or copy a Web page to another location, you must also move or copy the supporting folder in order to maintain all links to the Web page (p. 951). Courtner also teaches creating and adding web folders to a server (p. 873) for the Web pages, and other files. Courtner also teaches adding links to slides that can move the user from the current slide to another slide (p. 351). Compare to Claim 14, ***"... preparing and importing into a common page at said place a separate hypertext file for each said data object, together with any included images and support files, modifying said hypertext files to link to next and previous data objects within said document file; at read time, responsive to user selection of said next or previous control, issuing a uniform resource locator (url) to load and display a respective next or previous data object in sequence"***. It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Salas, Blumberg, and Courtner providing the benefit of having made a hypertext version of a presentation consisting of a number of slides and having made them available to collaborators using a browser.

In regard to dependent Claim 15, Salas fails to teach that *said data objects being presentation slides or spreadsheet worksheets*. However, Courtner teaches that when a file is saved as a web page, all supporting files, such as those for bullets, background textures, and graphics are saved in a separate subfolder. The subfolder is automatically named with the page name, followed by an underscore and the word files; for example, My Web Page_files. If the Web page is moved or copied to another location, the supporting folder must also be moved or copied in order to maintain all links to the Web page (p. 951). Courtner also teaches that spreadsheets can be saved as Web pages (pp. 1058-1061). Compare to Claim 15, ***"... said data objects being presentation slides or spreadsheet worksheets"***. It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Salas, Blumberg, and Courtner providing the benefit of having different ways of presenting data for use by collaborators.

In regard to dependent Claim 16, Salas fails to teach that *said presentation slides being Microsoft PowerPoint slides, said spreadsheet worksheets being Excel worksheets, said hypertext files being hypertext markup language (html) files*. However, Courtner teaches that PowerPoint presentation slides can be saved as Web pages (p. 291, AutoContent Wizard screens). Courtner also teaches that Excel spreadsheets can be saved as Web pages (pp. 1058-1061). Courtner also teaches saving documents as Web pages using the Save As Web Page option on the File menu (pp. 950-951 paragraphs 1-6). Compare to Claim 16, ***"... said presentation slides being Microsoft PowerPoint slides, said spreadsheet worksheets being Excel worksheets, said hypertext files being hypertext markup language (html) files"***). Courtner fails to teach *... said common page being a Notes document"*. However, one of ordinary skill in the art at the time of invention would have found that it was obvious that the common page was a Notes document providing the benefit of having bound the rest of the collaborative documents together into a common document.

Claim 18 is rejected under 35 U.S.C. 103(a) as being unpatentable over Salas in view of Blumberg.

In regard to independent Claim 18, Salas teaches in Fig. 7, that a user creates a new file or modifies a file downloaded from the server (14) (step 702). Once the user is finished editing the file, it may be uploaded to the server (14) to allow other users access to it. The user signals that the file should be transmitted to the server (14) by dragging the file onto an eRoom displayed by the browser (step 704). Dropping the file into the displayed eRoom invokes an ActiveX control or a background daemon process that manages the upload of the file to the server (14) (Col. 13, lines 14-26; compare with Claim 18, ***"... a browser interface window to said place including a new selector; a storage for storing said document file; an import scene; upload control; said import scene being responsive to user selection of said new selector and an import option for displaying said upload control; dragging and dropping said document file from said storage to said upload control"***). Salas fails to teach *said* *upload control being responsive to receiving said document file for, converting said document file to hypertext files; displaying a document file icon in said upload control; importing said hypertext files into said place*. However, Blumberg teaches that documents archived on a server computer can be accessed by client computers by means of web browsers. If a document is not in hypertext markup language (HTML) format, additional server-side or client-side processing may be necessary. Such server-side processing involves auxiliary software on the server for converting each document page requested into HTML format for viewing by the client (Col. 6, lines 1-7; compare to Claim 18, ***"... said upload control being responsive to receiving said document file for, converting said document file to hypertext files; displaying a document file icon in said upload control; importing said hypertext files into said place"***). It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Salas and Blumberg providing the benefit of having allowed an author to collaborate on a document with the option of having first previewed it from a browser.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to James H Blackwell whose telephone number is 703-305-0940. The examiner can normally be reached on Mon-Fri.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Joseph H Feild can be reached on 703-305-9792. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

JOSEPH FEILD
SUPERVISORY PATENT EXAMINER

James H. Blackwell
03/31/04